AMENDMENTS TO THE CLAIMS

This listing of claims replaces all prior versions in the application.

Listing of Claims:

CLAIM 1 (Currently Amended): A software method ~~virtual supercomputer apparatus~~ for constructing ~~comprising:~~ a reconfigurable virtual machine ~~(processor)~~, and both local and distributed networks of such virtual machines, each having its own instruction set, and operating on an underlying physical hardware processor or processors, comprising operations that enable:

~~or a local or distributed network of such virtual machines;~~

~~where~~

a) the virtual machine architecture ~~is designed to handle~~ to operate directly on ~~the~~ a class of problems having a solution describable in terms of nodes[[,]];

b) whereby a node comprises an index word and a data word~~, and;~~

c) whereby each node represents one or more of the following data structures ~~such as~~:

numeric tags, character tags, boolean flags, numeric values, character values, objects IDs, database-record IDs, simple arrays, variable-density multidimensional arrays, symbolic functions, mathematical functions, connection pointers to other nodes, function pointers, lookup-table list pointers, linked-lists, or even pointers to other solution spaces or data representations; ~~and~~

d) whereby nodes are interconnected in any of the following topologies ~~topology, such as~~:

independent point-clouds, ordered sets of points, acyclic graphs, cyclic graphs, balanced trees, recombining graphs, meshes, lattices, and various hybrids or combinations of such representations;

e) whereby ~~a~~ the virtual machine architecture includes ~~virtual hardware~~ units to:

configure nodes and virtual processor architecture, including register structures, node data structures, arithmetic modes, and memory schemes;

f) and virtual units to create (that is, instantiate) nodes, compute results for (that is, populate) nodes, move among (that is, navigate) nodes, and alter (that is, evolve) nodes and their interconnections;

g) ~~and~~ a virtual unit to provide highly-optimized function evaluation and fixed-point integer arithmetic, with application-selectable precision;

h) ~~and~~ a virtual unit to manage the distribution of data and processes to networked machines.

CLAIM 2 (Currently Amended): A ~~virtual supercomputing system, comprising:~~ software method that implements ~~the virtual machine of claim 1, and~~ a multi-tasking operating system for the virtual machine of Claim 1, that allows for multiprocessing via multiple virtual machines implemented on a network of underlying hardware processors, in a local or distributed cluster, comprising operations that:

a) creates a new virtual CPU for each task thread; ~~and~~

b) realize ~~contains:~~ software engines for configuring, instantiating, populating, navigating, and modifying (evolving) nodes;

c) realize autonomous daemons for background processing of nodes; ~~and~~

d) realize a toolbox containing frequently-used engine programs;

~~and allows for multiprocessing via multiple virtual machines implemented on a network of underlying hardware processors, in a local or distributed cluster;~~

e) realize an assembler for translating operating-system calls into virtual machine operation codes; ~~and~~

f) realize platform drivers for ~~implementing~~ virtual-machine operations on the underlying physical platform processor[[,]]; ~~via~~

g) realize a platform assembler for translating virtual machine operations into instruction-codes for platform operations.

CLAIM 3 (Currently Amended): A software method~~, or procedure, using the apparatus of Claim 1 and the system of Claim 2,~~ applied to the virtual machine of Claim 1, that allows for faster, and therefore lower cost, software application creation, and that produces computer programs that rapidly generate 'good enough' solutions to computationally complex and/or high-demand problems that are describable in terms of nodes, via a set of non-sequential processes, comprising:

a[[.]]) matching the virtual machine architecture and solution manifold to the problem architecture;

b[[.]]) adapting the solution manifold in response to changing demands in problem architecture or data;

c[[.]]) adapting the virtual-machine architecture in response to changing demands in problem architecture or data;

d[[.]]) using application-selectable arithmetic precision to rapidly compute 'accurate enough' calculations when evaluating nodes;

e[[.]]) using software-emulation of supercomputing techniques, such as small instruction set, simple and efficient data representation and handling, inherent vector representation, limited data/calculation modes, interleaved memory, table lookup, induced pointers, and distributed & parallelized computation;

f[[.]]) separating the populating and navigating of nodes, to allow for pre-computation of manifolds, so that navigation of possible solutions occurs in near real-time;

g[[.]]) using autonomous, second-order dedicated processes that operate in background, as concurrent tasks, to collect garbage, prune trees, condense redundancies, process edit-queues, interpolate with finer granularity (mesh enhancement) around selected nodes in state-space, or to extrapolate and elaborate the data structures, during both population and navigation phases;

h[[.]]) generating virtual CPUs for each operating-system task thread.